3GPP TSG-RAN WG2 #118-e R2-2206708

Electronic meeting, 2022-05-09 - 2022-05-20

Agenda Item: 6.18.1

Source: Ericsson

Title: Phase2, CR review - Report from [AT118-e][507][RA Part] CP open issues and CR 38.331 (Ericsson)

Document for: Discussion, Decision

# 1 Introduction

This contribution is a summary of the discussion for the remaining CP issues /RILs for RICS (RA Partitioning):

* [AT118-e][507][RA Part] CP open issues and CR 38.331 (Ericsson)

CP open issues and CR capturing agreed corrections

Deadline: To be set by rapporteur aiming to have company inputs and proposals by

**The following agreements were reached in the first and second online session:**

Agreement

1. Use SetupRelease-structure, similar to the legacy RACH config. And call the field/IEs "list" as they provide a list of additional RACH configurations. Update IE name accordingly
2. Delete the extension marker and the field laterThanRel17Features from FeatureCombination IE and use spare fields for future extendibility. FFS the number of spare values
3. Add a non-critical extension (i.e., extension marker) in the FeatureCombinationPreambles IE
4. Add msgA-RSRP-Threshold (without SSB suffix) in partition
5. Allow partition-specific msgA PUSCH resources. If not provided we use the general PUSCH
6. rsrp-ThresholdMsg3 is put in BWP-UplinkCommon, editor’s note is removed, and field description is added.
7. FFS pending slicing discussion - add fields feature-RA-PrioritizationForAccessIdentity-r17 and ra-PrioritizationForAccessIdentity-r16 and verify if it is clear how the UE selects. Ask question in email discussion for other non-slicing features
8. Change the name of the field "featureCombinationPreambles" to "featureCombinationPreamblesList"

**Second online session Agreements:**

Agreements:

1   Adopt H537 with the additional correction.

2   Implement H902 with the addition of “and all RACH configurations” in field description.

3   Implement H904 with the addition from HW. Discuss additional changes for legacy text with CR

4  Adopt the text proposal using 4 spares in FeautureCombination IE.

5 dd extension markers in FeatureCombinationPreambles (outside wrapper sequence of featureSpecificParameters).

6 Adopt the proposal in H904 to capture that the field is mandatory if there are both 2-step and 4-step RA resources for a particular feature combination in a BWP.

7 No additions or changes are introduced for signalling which configuration to use when fallback from CFRA in 38.331, at this point based on the previous agreement.

8 Adopt the text proposals for rsrp-ThresholdMsg3 in BWP-UplinkCommon above with the editorial correction “The field is mandatory if both set(s) of Random Access resources with MSG3 repetition indication and set(s) of Random Access resources without MSG3 repetition indication are configured in the BWP. It is absent otherwise”.

9 The IE feature-RA-PrioritizationForAccessIdentity is not added for general RA partitioning

10 As a baseline no RACH partitioning specific capability is defined

## 2.8 Other remaining correction proposals

**The following topics were proposed for discussion and left for the CR review phase:**

🡪 Note that the items below are summarized and as such may need some adjustment or additions should they be agreed to be corrected. Please refer to the source document if avaliable for more details.

**In R2-2206127 The following corrections were proposed:**

1. **The usage and the absence of the *ssb-SharedRO-MaskIndex-r17* field should be interpreted differently depending on where *FeatureCombinationPreambles* IE is configured.**

With the following change to 38.331, other fields omitted :

|  |
| --- |
| ***FeatureCombinationPreambles field descriptions*** |
| ***ssb-SharedRO-MaskIndex***  Mask index (see 38.321).  Indicates a subset of ROs where preambles are allocated for this feature combination. If this field is configured within *FeatureCombinationPreambles* which is included in *RACH-ConfigCommon*, it indicates a subset of ROs configured within this *RACH-ConfigCommon.* If this field is configured within *FeatureCombinationPreambles* which is included in *RACH-ConfigCommonTwoStepRA*, in case of separate ROs for 4-step and 2-step random access, this field indicates a subset of ROs configured for 2-step random access, and in case of shared ROs, it indicates the subset of ROs configured for 4-step random access. This field can only be configured when there is more than one RO per SSB. If the field is absent, for RACH associated to a feature combination indicated by *featureCombination*, the UE utilizes the ROs as configured by the *RACH-ConfigCommon* or *RACH-ConfigCommonTwoStepRA* in which this *FeatureCombinationPreambles* is included. |

**Rapporteur comment:**

As an initial assessment this does not seem to be needed and comes with additional issues.

Yellow marked: Already clear as the configuration already determines which ROs are used for 2-step and for 4-step.

Red marked: This addition seems misleading and may lead to errors. The exiting text and the signaling structure as it is now, seems to be clear.

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | We agree that the some clarification is needed for the absent case. However, we also agree with the rapporteur’s comment that the red part would imply additional restriction.  Since there are too many cases of configuring *FeatureCombinationPreambles* in *RACH-ConfigCommon* or *RACH-ConfigCommonTwoStepRA* (may be included in legacy RACH configuration or *additionalRACH-ConfigCommon*), it is not reasonable to specify all cases of configuration. Therefore, following modification is enough to simply clarify that it shares all ROs of RACH configuration that contains this field, when the field is absent:  ‘…If the field is absent, all ROs configured in *RACH-ConfigCommon* and *RACH-ConfigCommonTwoStepRA* containing this *FeatureCombinationPreambles* are shared.’ |
| Huawei, HiSilicon | We do not agree the field description is clear. At the moment it just says that it indicates “a subset of ROs.” However, it is completely unclear which ROs are referred to. In legacy configuration it was clear because 2-step RA ROs could only be a subset of a single 4-step RA ROs. Now the subset can refer to:  1. 4-step ROs for a feature combination being a subset of legacy 4-step ROs.  2. 4-step ROs for a feature combination being a subset of 4-step ROs configured in additional RACH config.  3. 2-step ROs for a feature combination being a subset of 2-step ROs configured in additional RACH config.  4. 2-step ROs for a feature combination being a subset of 4-step ROs configured in additional RACH config.  And in case the field is absent, the current field description all ROs are shared, which is confusing as certainly the intention is not to share all ROs from all RACH configurations. In our understanding the intention is only to share the ROs in the additional RACH configuration where the feature combination is configured. We tried to clarify by the color coding how this relates to the updated field description:  ***ssb-SharedRO-MaskIndex***  Mask index (see 38.321).  Indicates a subset of ROs where preambles are allocated for this feature combination. If this field is configured within *FeatureCombinationPreambles* which is included in *RACH-ConfigCommon*, it indicates a subset of ROs configured within this *RACH-ConfigCommon.* If this field is configured within *FeatureCombinationPreambles* which is included in *RACH-ConfigCommonTwoStepRA*, in case of separate ROs for 4-step and 2-step random access, this field indicates a subset of ROs configured for 2-step random access, and in case of shared ROs, it indicates the subset of ROs configured for 4-step random access. This field can only be configured when there is more than one RO per SSB. If the field is absent, for RACH associated to a feature combination indicated by *featureCombination*, the UE utilizes the ROs as configured by the *RACH-ConfigCommon* or *RACH-ConfigCommonTwoStepRA* in which this *FeatureCombinationPreambles* is included. |
| Xiaomi | We agree the yellow marked with following modification to align with the 4-step RA case.  “If this field is configured within *FeatureCombinationPreambles* which is included in *RACH-ConfigCommonTwoStepRA*, it indicates a subset of ROs configured within this *RACH-ConfigCommonTwoStepRA*. ”  For the absent case, we prefer to adopt the LGE’s modification with minor changes:  ‘…If the field is absent, all ROs configured in *RACH-ConfigCommon* or *RACH-ConfigCommonTwoStepRA* containing this *FeatureCombinationPreambles* are shared.’ |
|  |  |

*Summary: The rapporteur adopted Xiaomis proposal as it aligns nicely as a middle ground between the proposals. However, there has been limited input and time, the rapporteur thinks this can be corrected (if wrong) based on input in the next meeting. WIth this change, the need code is changed from "Need R" to "Need S".*

1. **The group B related parameters in *FeatureCombinationPreambles* should be grouped together and it should be clarified that when these parameter are not provided in *FeatureCombinationPreambles*, the UE should consider there is only one preamble group configured for the RACH resources for the associated feature combination.**

With the following proposed correction in ***FeatureCombinationPreambles***:

***FeatureCombinationPreambles* information element**

-- ASN1START

-- TAG-FEATURECOMBINATIONPREAMBLES-START

FeatureCombinationPreambles-r17 ::= SEQUENCE {

featureCombination-r17 FeatureCombination-r17,

startPreambleForThisPartition-r17 INTEGER (1..64),

numberOfPreamblesForThisPartition-r17 INTEGER (1..64),

ssb-SharedRO-MaskIndex-r17 INTEGER (1..15) OPTIONAL, -- Need S

groupBconfigured-r17 SEQUENCE {

ra-Msg3SizeGroupA ENUMERATED {b56, b144, b208, b256, b282, b480, b640,

b800, b1000, b72, spare6, spare5,spare4, spare3, spare2, spare1},

messagePowerOffsetGroupB ENUMERATED { minusinfinity, dB0, dB5, dB8, dB10, dB12, dB15, dB18},

numberOfRA-PreamblesGroupA INTEGER (1..64)

} OPTIONAL, -- Need S

separateMsgA-PUSCH-Config-r17 MsgA-PUSCH-Config-r16 OPTIONAL, -- Cond MsgAConfigCommon

featureSpecificParameters-r17 SEQUENCE {

rsrp-ThresholdSSB-r17 RSRP-Range OPTIONAL, -- Need R

rsrp-ThresholdMsg3-r17 RSRP-Range OPTIONAL, -- Need R

-- Editor's note: TBD if this parameter indeed can be partition-specific.

deltaPreamble-r17 INTEGER (-1..6) OPTIONAL -- Need R

}

}

-- TAG-FEATURECOMBINATIONPREAMBLES-STOP

-- ASN1STOP

|  |
| --- |
| ***groupBconfigured***  Preamble grouping for the feature combination. If the field is absent then there is only one preamble group configured. |

**Rapporteur comment:**

Proposal should be accepted as this follows the legacy signaling for when groupB is configured and their grouping in signaling structure.

Perhaps a modification if absent: “If absent only groupA configured”

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | Agree with the proposal. We have no strong view on the rapporteur’s modification, i.e., can follow either way. |
| Huawei, HiSilicon | No strong view on the proposed modification from the rapporteur (but OK). |
| Xiaomi | Agree with the proposal, and fine with the rapporteur’s modification. |
| ZTE | Agree |

*Summary: The rapporteur adopted the proposal without changes.*

1. **When the *separateMsgA-PUSCH-Config* is not provided in *FeatureCombinationPreambles*, UE should apply the corresponding parameter in the *RACH-ConfigCommonTwoStepRA* of the BWP which includes the *FeatureCombinationPreambles* IE.**

With the following proposed correction in ***FeatureCombinationPreambles* field descriptions**:

|  |
| --- |
| ***separateMsgA-PUSCH-Config***  If present it specifies how the 2-step RACH preambles identified by this *FeatureCombinationPreambles* are mapped to a PUSCH slot separate from the one defined in MsgA-ConfigCommon-r16. If the field is absent, the UE should apply the corresponding parameter in the *RACH-ConfigCommonTwoStepRA* of the BWP which includes the *FeatureCombinationPreambles IE*. |

**Rapporteur comment:**

Proposal should be accepted.

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | Agree with the proposal. |
| Xiaomi | Agree with this proposal. |
| ZTE | Agree |

*Summary: The rapporteur adopted the proposal without changes.*

1. **The condition description of the condition “2StepOnly” needs to be updated to clarify that the parameters are mandatory in case *AdditionalRACH-ConfigCommon* does not contain 4-step RA configuration.**

With the following correction proposal in ***RACH-ConfigCommonTwoStepRA* field descriptions**:

|  |  |
| --- | --- |
| *2StepOnly* | The field is mandatory present in *msgA-ConfigCommon* fieldin *BWP-UplinkCommon* if *rach-ConfigCommon* field is absent in this *BWP-UplinkCommon*, otherwise the field is optionally present in *msgA-ConfigCommon* fieldin *BWP-UplinkCommon*, Need S.  The field is mandatory present in *msgA-ConfigCommon* fieldin *AdditionalRACH-ConfigCommon* if *rach-ConfigCommon* field is absent in this *AdditionalRACH-ConfigCommon,* otherwise the field is optionally present in *msgA-ConfigCommon* fieldin *AdditionalRACH-ConfigCommon*, Need S. |

**Rapporteur comment:**

Proposal should be accepted as this aligns to the current condition and applies to when there is an additionalRACH-CommonConfig configured.

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | No strong view.  In our understanding, if this is implemented, some parameters (e.g., msgA-TransMax-r16, msgA-RestrictedSetConfig-r16) shall be configured per ‘AddtionalRACH-ConfigCommon,’ which may cause the signaling overhead. However, considering it’s too late to introduce new condition, we can accept this if the majority supports.  By the way, if this is implemented, we think that the same change should be implemented the condition description of the “2StepOnly” included in *RACH-ConfigGenericTwoStepRA* field description. |
| Huawei, HiSilicon | As clarified by the rapporteur, the intention is to make the condition applicable to additional RACH configurations. The description field is misleading and incomplete otherwise. |
| Xiaomi | Fine with this clarification. |
| ZTE | The proposal seems fine. |

*Summary: The rapporteur adopted the proposal without changes.*

1. **The 4-step RACH configuration and the 2-step RACH configurations of one feature combination should be provided in the same *AdditionalRACH-ConfigCommon*.**

With the following correction proposal in ***BWP-UplinkCommon* field descriptions**:

|  |
| --- |
| ***additionalRACH-ConfigCommon***  List of feature or feature combination-specific RACH configurations, i.e. the RACH configurations configured in addition to the one configured by *rach-ConfigCommon* and by *msgA-ConfigCommon*. The *rach-ConfigCommon* and *msgA-ConfigCommon* for one feature or feature combination should be provided in the same *additionalRACH-ConfigCommon*. |

**Rapporteur comment:**

Proposal should not be implemented as current signaling allow all 4-step preambles in one Additional (4-step only) RACH config, and all 2-step preambles in one Additional (2-step only) RACH config. We can also have an AdditionalRACH-Config common that has both 2-step and 4-step. The proposal thus introduces a restriction that has not been agreed or discussed in detail.

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | No strong view. |
| Huawei, HiSilicon | This topic was discussed in our paper in R2-2206127 and it was also submitted as RIL A022. As calrified in R2-2206127, the reason for such change is that it was agreed that the UE should use parameters from 4-step RA configuration of the same feature combination in case the the corresponding parameter is not provided explicitly in 2-step RA configuration. In case 4-step RA configuration is now placed in a different additional RACH configuration than 2-step RACH configuration, this becomes very complex for the UE as it needs to browse through all the RACH configurations to get some parameter values. At the same time, it is unclear what the benefit is of keeping the flexibility mentioned by the rapporteur. |
| Xiaomi | No strong view. |
| ZTE | We don’t think such restriction is needed. If majority companies want to have this, we prefer to have an exception for the legacy RO, since one feature specific RACH partition may share the RO of legacy 4-step RACH resource but have separate RO in additionalRACH-ConfigCommon for 2-step RACH resource (or vice versa). Such flexibility should be allowed.  If rach-ConfigCommon for one feature or feature combination is provided, the corresponding msgA-ConfigCommon for the same feature or feature combination can only be included in the same additionalRACH-ConfigCommon or in the msgA-ConfigCommon within BWP-UplinkCommon |

Summary: There are different views, the rapporteur did not implement this change. Can be discussed further in the next meeting.

1. **The condition description of the condition “2Step4Step” need to be updated to clarify that the field is mandatory present in *msgA-ConfigCommon* field in *AdditionalRACH-ConfigCommon* if both 2-step random access type and 4-step random access type are configured for the same feature combination in the BWP.**

With the following correction proposal in ***RACH-ConfigCommonTwoStepRA***:

|  |  |
| --- | --- |
| *2Step4Step* | The field is mandatory present if both 2-step random access type and 4-step random access type are configured in the BWP, otherwise the field is not present.  The field is mandatory present in *msgA-ConfigCommon* field in *AdditionalRACH-ConfigCommon* if both 2-step random access type and 4-step random access type are configured for the same feature combination in the BWP. |

**Rapporteur comment:**

Proposal should be implemented since this clarifies the legacy description but for the case of additional RACH config common (i.e. not per BWP).

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | Agree with the proposal and the rapporteur’s comment. |
| Xiaomi | Agree with the proposal and the rapporteur’s comment. |
| ZTE | Agree with the intention.  Some clarification may be useful for the first sentence to clarify that it is only applicable for no feature combination case. If not, the “otherwise clause” in the first sentence seems to be confusing (i.e. it seems to say that the field is not present in all other cases, but then we have some further condition below that says the field is present in case of some feature combination specific use case). |

*Summary: The proposal is adopted. If some modifications are needed, they can be discussed in the next meeting.*

**In comments to remaining issues it was brought forward the following items:**

**7)**

|  |  |
| --- | --- |
| ZTE | As noted in Z379, it seems that the current signalling structure allows additional RACH resources which are not associated with any feature combination. However, it is not clear to us if this is the intention and MAC spec assumes that there is one set of RACH resources that are feature combination agnostic (i.e. the legacy RACH resources). So, it is worth clarifying that network will always associate the additional RACH resources to some feature combination. |

**Rapporteur comment:**

This depends on if in 38.321 MAC, it is clear that the case of un-associated additional RACH resource to feature combination is handled or not. If not, a clarifying sentence may be needed.

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | In our understading, a set of RA resources not associcated with any feature (i.e., legacy RACH resource) should only be configured by legacy signaling (i.e., no set of RA resource which is not associated with any feature in additionalRACH-ConfigCommon). It is specified in the field description of additionalRACH-ConfigCommon as follows:   |  | | --- | | *BWP-UplinkCommon* field descriptions | | ***additionalRACH-ConfigCommon***  List of feature or feature combination-specific RACH configurations, i.e. the RACH configurations configured in addition to the one configured by *rach-ConfigCommon* and by *msgA-ConfigCommon*.  *Editor's note: Naming of this can be discussed further, e.g. to make it clear that this field can configure msgA-ConfigCommons also* |   If the additional description is needed, the condition description can be added to featureCombinationPreambles IE under the RACH-ConfigCommon or RACH-ConfigCommonTwoStepRA in order to ensure that the additional RACH configuration should be associated with feature/feature combination. |
| Huawei, HiSilicon | Agree with LGE. We can handle this by adding a condition that FeatureCombinationPreambles is mandatory if provided in AdditionalRACH-ConfigCommon. |
| ZTE | As pointed out by other companies, only one set of legacy RACH resources that are not associated with any feature combination is supported by MAC. So, we should have some restriction in RRC to avoid more than one such set.  Even with the FeatureCombinationPreambles being mandatory, there are still some preambles/RA resource which is not associated to any feature combination (i.e. the RA resource defined by the “ssb-perRACH-OccasionAndCB-PreamblesPerSSB” and “groupBconfigured” (if configured) within the rach-ConfigCommon-r17 but not in featureCombinationPreambles, which is similar as the one configured as legacy RACH resource). We need to clarify these resource shall not be used by UE.  We can either clarify this in field description or have a new structure which include ssb-perRACH-Occasion only (without CB-PreamblesPerSSB) and include “groupBconfigured” only in featureCombinationPreambles. |

*Summary: Adopted the proposal as follows in BWP-UplinkCommon. Can be discussed further if this is not sufficient.*

|  |
| --- |
| ***additionalRACH-ConfigCommonList***  List of feature or feature combination-specific RACH configurations, i.e. the RACH configurations configured in addition to the one configured by *rach-ConfigCommon* and by *msgA-ConfigCommon*. The network associates all possible preambles of an additional RACH configuration to a feature or feature combination. |

8)

|  |  |  |
| --- | --- | --- |
| LGE | In the current signaling stucture of *FeatureCombination* IE, the Rel-17 UE bahaviour with the spare fields is very unclear. Unless the UE operation is explicitly specified, the decoding of spare fields would be handled by general RRC error handling (as defined in Clause 10 of TS 38.331). However, as defined in clause 10.5 of TS 38.331, the Rel-17 UE will ignore the values of the spare fields which are not readable.   |  | | --- | | 10.5 Not comprehended field The UE shall, when receiving an RRC message on any logical channel:  1> if the message includes a field that the UE does not comprehend:  2> treat the rest of the message as if the field was absent. |   In this case, the Rel-17 UEs may use the RA resource for the potential feature, even though it is not allowed. For example, when NewFeature is defined in later release, Rel-17 RedCap UE shall not use the RA partition associated with following FeatureCombination:  FeatureCombination-r17 ::= SEQUENCE {  redCap-r17 {true}  NewFeature(spare4) {true}  }  However, since the Rel-17 RedCap UE cannot comprehend the indication of NewFeature, the Rel-17 RedCap UE treat the FeatureCombination IE as if the field(i.e., NewFeature) is absent. Therefore, the Rel-17 RedCap UE would use the RACH partition associated with this FeatureCombination, which is wrong operation.  In our understanding, if one of spare fields is set, Rel-17 UE shall NOT use that partition even though the other feature combinations are matched to the Rel-17 UE. Therefore, in order to clarify the UE behavior with undefined spare fields, one of the followings is needed:   * Option 1: leave the laterThanRel17Features field in order to disable the RACH resource configured for potential features. * Option 2: Specify the Rel-17 UE bahaviour when at least one of the spare fields is set to {true}.   In our view, since the exact definition of spare fields would not be determined in Rel-17, it is very hard to specify the UE operation with undefined field. Therefore, it is proposed to leave laterThanRel17Features field in the *FeatureCombination* IE (i.e., Option 1). |

**Rapporteur comment:**

Handling of spares may need an additional sentence to clarify the Rel-17 UE bahaviour when at least one of the spare fields is set to {true}. The handling of resource selection is made in MAC, so this should be clear so that MAC does not consider a partition associated by features not comprehended by the UE. RAN2 to discuss how to implement if something is considered needed.

**Company input:**

|  |  |
| --- | --- |
| **Company** | **Comments** |
| LGE | We think additional clarification is essential since current specification (including MAC and RRC) cannot decribe the Rel-17 UE behavior for Rel-18 feature.  In the current MAC spec, the availability of set of RA resource is determined as follows:  - If FeatureX indication is configured for a set of Random Access resource, the set of Random Access resources as not available for the Random Access procedure if FeatureX is not applicable  However, as described in 10.1 of TS 38.331, the spare field would be considered as a not comprehended field. Therefore, when the UE receives the FeatureCombinaiton IE with the feature indication for new feature (i.e., spare fields in Rel-17), the Rel-17 UE ignores feature indication of the new feature and treat the rest of FeatureCombination IE. Therefore, the MAC entity would not be able to determine whether the feature indication for new feature is configured or not, and the availablilty of the set of RA resource is determined except the the feature indication for new features, which is wrong operation.  In our view, it would be better to clarify the UE behavior in TS 38.331 rather than TS 38.321, since it is related to encoding method of RRC message. We are fine with any description as long as the UE behavior is clear enough(i.e., Rel-17 UE shall not use the RACH partition when at least one of the spare fields is set to {true}), and the corresponding text could be added in the Field definition of FeatureCombination IE. However, the definition of spare fields would not be determined in Rel-17, and it is hard to specify the UE behavior with uncomprehended field (i.e., UE would not determine whether this uncomprehended field is set to {true}).  Therefore, we support the simple approach for this, which is to define a Rel-17 UE operation with the field defined in Rel-17. That is, leave an indication for Rel-17 UEs in order to disable the RA resource associated with the features in future release (i.e., laterThanRel17Features field). |
| Huawei, HiSilicon | We support Option 2 from LGE’s issue description. Perhaps, as the rapporteur suggests, it is best placed in MAC specifications. |
| LGE2 | If majority supports Option 2, we are fine to go with this way as long as the UE behavior is clear enough(i.e., Rel-17 UE shall not use the RACH partition when at least one of the spare fields is set to {true}).  For example, following text can be added in the definition of *FeatureCombination* IE:   |  | | --- | | – *FeatureCombination* The IE *FeatureCombination* indicates a combination of features to be associated with a RA partition (i.e. an instance of *FeatureCombinationPreambles*). In this release, if at least one of the spare fields is set to {true}, the UE shall not use the associated *featureCombinationPreambles*. | |
| Xiaomi | Prefer option2. And agree with rapporteur and HW that it is better to handle in MAC spec. |
| ZTE | We are fine with either option 1 and option 2. For option 2, we prefer to clarify this in RRC that the UE should ignore the RACH resource associated to the feature combination if any feature within the feature combination is not supported or unknown |

Summary: Adopted as follows in FeatureCombinationPreambles-r17. Can discuss further.

|  |
| --- |
| ***featureCombination***  Indicates which combination of features that the preambles indicated by this IE are associated with. The UE ignores a RACH resource defined by this *FeatureCombinationPreambles* if any feature within the *featureCombination* is not supported by the UE or has an unknown value. |

9)

|  |  |
| --- | --- |
| Apple | Related to RIL A022. Same comment as proposal 5 from Huawei.  Current sepc is unclear on the RACH resource assocation between 2-step RACH and 4-step RACH for fallback case, and it worth some clarification.  **Proposal 3: Confirm the 2-step RACH configuration and 4-step RACH configuration for fallback case are from the same *FeaturecombinationPreamble* and the same *AdditionalRACH-ConfigCommon*.**  In addition, the link between the selected set of the Random Access resources in MAC spec (section 5.1.1b) and the RRC configuration (i.e*. AdditionalRACH-ConfigCommon, and FeaturecombinationPreamble)* is lack, and we should make it clear in the spec. |

**Rapporteur comment:**

See 5) and if remaining issues this can be discussed further based on consensus.